УДК 004.931, 519.174

А.В. Агарков
Институт проблем искусственного интеллекта
МОН Украины и НАН Украины, г. Донецк
Украина, 83048, г. Донецк, ул. Артема, 118-б

Поиск объектов на изображениях с помощью графов

A.V. Agarkov
Institute of Artificial Intelligence
MES of Ukraine and NAS of Ukraine, c. Donetsk
Ukraine, 83048, c. Donetsk, Artema st., 118-b

Object Searching in Images by Graph Using

А.В. Агарков
Інститут проблем штучного інтелекту
МОН України і НАН України, м. Донецьк
Україна, 83048, м. Донецьк, вул. Артема 118-б

Пошук об’єктів на зображеннях за допомогою графів

Данная работа посвящена проблеме поиска объектов на изображении. Для её решения предлагается использовать метод на основе применения графа, описывающего изображение. Данный граф должен обладать тем свойством, что отдельным объектам на изображении должны соответствовать его подграфы. Тогда задача поиска объекта сводится к выделению соответствующего подграфа. Для описания изображения предлагается использовать граф, построенный на основе использования структурных элементов, выделенных с помощью мультимасштабного представления. Данный граф позволяет отражать структуру как текстурированных, так и нетекстурированных объектов. Для выделения подграфов, соответствующих искомым объектам, используются средства, основанные на построении дополнительного графа-пирамиды. Экспериментальная проверка использования данного графа и средств его обработки для поиска текстурированных и слабо текстурированных объектов показала их высокую эффективность. Сравнение с аналогами показало значительное превосходство предложенных средств при поиске слабо текстурированных объектов.
Ключевые слова: поиск объектов, распознавание изображений, сравнение графов.

This work is devoted to object search in images. To solve this problem, the method based on the use of a graph that describes the image is applied. The given graph should have such property that the individual objects in the image must comply with its subgraphs. In such case, the task of object search is limited to allocation of corresponding subgraph. To describe the images, it is suggested to use a graph constructed on the basis of the structural elements, which are identified with the help of multi-scale representation. This graph can reflect the structure of both textured and non-textured objects. To separate subgraphs, which correspond to the desired objects, the tools based on the construction of additional graph pyramid are used. Experimental verification of a given graph, and means of processing to search for textured and weakly textured objects demonstrated their high efficiency. Comparison to analogues showed significant superiority of the proposed means at searching poorly textured objects.
Key Words: object search, image recognition, graph comparison.

Дана робота присвячена проблемі пошуку об’єктів на зображенні. Для її вирішення пропонується використовувати метод на основі вживання графа, що описує зображення. Даний граф повинен володіти тією властивістю, що окремим об’єктам на зображенні повинні відповідати його підграфи. Тоді задача пошуку об’єкта зводиться до виділення відповідного підграфа. Для опису зображення пропонується використовувати граф, побудований на основі використання структурних елементів, виділених за допомогою мультимасштабного представлення. Даний граф дозволяє відображати структуру як текстурованих, так і нетекстурованих об’єктів. Для виділення підграфів, відповідних шуканим об’єктам, використовуються засоби, засновані на побудові додаткового графа-піраміди. Експериментальна перевірка використання даного графа і засобів його обробки для пошуку текстурованих і слабо текстурованих об’єктів показала їх високу ефективність. Порівняння з аналогами показало значну перевагу запропонованих засобів при пошуку слабо текстурованих об’єктів.
Ключові слова: пошук об’єктів, розпізнавання зображень, порівняння графів.

Введение

В последнее время для поиска / распознавания произвольных объектов на изображениях всё чаще применяются методы, использующие локальные инвариантные черты (local invariant features) изображений. Данные черты довольно стабильно выделяются на изображениях даже при смене освещения, масштаба и ракурса. Это позволяет использовать их для описания и поиска / распознавания довольно широкого класса объектов на изображениях. Каждой характерной черте ставится в соответствие дескриптор, который отражает особенности изображения в её окрестности. При проведении поиска / распознавания на рассматриваемом изображении выделяются все локальные инвариантные черты, затем среди них проводится поиск множества тех, которые соответствуют искомому объекту. Для этого вначале для каждой черты модели проводится поиск наиболее вероятных соответствий из черт рассматриваемого изображения путём сравнения соответствующих дескрипторов. Чтобы отфильтровать ложные соответствия используется также метод FLANN, который учитывает и соседние черты.

После выделения наиболее вероятных соответствий для отдельных черт модели проводится поиск соответствия их множеству среди множества черт рассматриваемого изображения. Наиболее распространенными методами для поиска соответствующего объекту множества локальных черт в настоящее время являются RANSAC [1] и обобщённое преобразование Хафа.

Данная схема в настоящее время широко и успешно используется с различными вариациями – применяются различные детекторы локальных черт, различные дескрипторы и способы их сравнения [2-4]. Однако следует заметить, что данный метод / схема может успешно применяться только для поиска / распознавания объекта по имеющемуся шаблону и искомый объект должен быть достаточно текстурирован (т.е. иметь внутреннюю структуру). Это связано с особенностями применяемых дескрипторов и методов поиска соответствующих множеств черт. Поскольку при поиске соответствующих подмножеств локальных черт используется их взаимное расположение, то логично для их описания использовать граф. Вершинам данного графа-описания соответствуют структурные элементы (локальные черты), а ребра отражают отношения между ними.

Отдельным объектам на изображении соответствуют подграфы графа-описания. Таким образом, задача поиска определённого объекта сводится к поиску соответствующего подграфа, который осуществляется путем сравнения графа-эталона с графом-описанием изображения.

Данный подход был использован в работах [5], [6], где в качестве структурных элементов использовались границы. Использование в качестве структурных элементов границ приносит определённые трудности, связанные с влиянием изменений яркости и контрастности на результат их выделения. Гораздо предпочтительней для этого локальные инвариантные черты, хотя их недостатки, упомянутые выше, несколько ограничивают область их применения.

В данной работе предлагается усовершенствование данного метода за счёт использования новых графов и средств их сравнения. В качестве графа для сравнения используется клика, построенная на основе графа-описания, предложенного в [7-9]. Использование данного графа-описания в совокупности со сравнением графов, основанным на построении дополнительного графа-пирамиды [10], [11], позволяет проводить поиск / распознавание объектов на изображениях без использования сложных дескрипторов. Также предложенное усовершенствование позволяет проводить поиск одинаковых объектов на различных изображениях без наличия априорной информации о них.

Целью данной работы является проверка эффективности применения графа-описания и средств его обработки, предложенных в работах [7-11], для решения задач распознавания зрительных образов. Для этого необходимо:

1. Разработать алгоритм на основе их применения для поиска текстурированных и нетекстурированных объектов на изображениях;
2. Провести проверку эффективности работы данного алгоритма, сравнить эффективность работы данного алгоритма с аналогами.

Граф-описание изображения

Выделение структурных элементов для построения графа-описания описано в работах [7-9] и является развитием метода выявления структуры изображений, основанного на применении мультимасштабных представлений, получаемых на основе свёрток с функцией Гаусса и её производными. Данные структурные элементы, называемые далее инвариантными, так же как и инвариантные локальные черты, выделяются с помощью мультимасштабного представления, построенного с помощью оператора DoG.

Инвариантные структурные элементы (ИСЭ) выделяются на основе использования экстремумов по направлению во всех плоскостях мультимасштабного представления рассматриваемого изображения. Следует отметить, что множество локальных инвариантных черт является подмножеством множества ИСЭ. Это связано с тем, что локальные экстремумы, которым соответствуют локальные инвариантные черты, являются подмножеством экстремумов по направлению, используемых для выделения инвариантных структурных элементов. И, так же как и локальные черты, ИСЭ характеризуются координатами центра, характерными масштабом и яркостью.

Таким образом, в качестве графа, описывающего изображение I, используется граф $G_I(V_I, E_I)$, где V_I и E_I – множества вершин и рёбер соответственно. Каждой вершине $v_i^I \in V_I$ взаимно однозначно соответствует структурный элемент, который характеризуется координатами центра $\vec{r}(v_i^I) = (x, y)$, характерным масштабом $S(v_i^I)$, характерной яркостью $G(v_i^I)$, контрастностью $DoG(v_i^I)$. Параметры, характеризующие структурный элемент, составляют атрибут соответствующей вершины. Каждое ребро $e_{ij}^I = e(v_i^I, v_j^I) \in E_I$ характеризует взаимное расположение структурных элементов, отвечающих вершинам. Атрибутом ребра являются расстояние между центрами соответствующих структурных элементов и отношения их характерных масштабов, яркостей и контрастностей:

$$RS(e_{ij}^I) = S(v_i^I) / S(v_j^I),$$

$$RG(e_{ij}^I) = G(v_i^I) / G(v_j^I),$$

$$RDoG(e_{ij}^I) = DoG(v_i^I) / DoG(v_j^I),$$

$$RGDoG(e_{ij}^I) = \frac{G(v_j^I) * DoG(v_i^I)}{DoG(v_j^I) * G(v_i^I)}.$$

Для ускорения сравнения изображений целесообразно использовать не все инвариантные структурные элементы, а только часть. То есть необходимо выбрать репрезентативные вершины, которые будут участвовать в сравнении графов-описаний и которые образуют репрезентативный граф. Данный граф удобнее всего делать кликой, чтобы отсутствие каких-либо вершин не влияло на результат поиска соответствующих подграфов.

Таким образом, алгоритм поиска одинаковых объектов на изображениях имеет вид:
1. Построение графов-описаний сравниваемых изображений.
2. Отбор репрезентативных вершин для построения графов-клик, которые будут сопоставляться.

3. Построение производного графа.
4. Поиск в производном графе множества максимальных клик
5. Формирование решения из найденных клик.

Если в качестве одного из сравниваемых изображений будет изображение объекта, то данный алгоритм будет применим для поиска определённого объекта на изображении.

Репрезентативный граф

Чаще всего в качестве репрезентативных вершин достаточно выбрать те, которым соответствуют ИСЭ, совпадающие с локальными инвариантными чертами. Каждой репрезентативной вершине v_i^R ставится в соответствие множество ИСЭ $b(v_i^R)$, которые расположены в окрестности соответствующей ЛИЧ и характеризуют структуру изображения в данной окрестности.

$$b(v_i^R)=\left\{v_j \in V_I : r(v_i^R,v_j)\le S(v_i^R)*k, \exists C(v_i^R,v_j)\right\}$$

$C(v_i^R,v_j)$ – цепь между вершинами v_i^R и v_j $C(v_i^R,v_j)=\left\{v,e : v\in b(v_i^R), m(e)=1\right\}$,

где $m(e)=1$ означает, что соседние структурные элементы соответствующие вершинам, инцидентные ребру e, относятся к одной детали изображения и их центры являются соседними пикселями ([8]).

Алгоритм для формирования множества $b(v_i^R)$ имеет вид

$b(v_i^R)=\varnothing$ // инициализация искомого множества $b(v_i^R)$ пустым множеством

$b(v_i^R)+=v_i^R$ // добавление в искомое множество вершины v_i^R

$for(j=1; j\le N\left(b(v_i^R)\right); j++)$

{

$//v_{i,j}^R - j - ая\, вершина\, из\, множества\, b(v_i^R)$

$//\ n(v_{i,j}^R) - множество\, вершин, смежных\, с\, v_{i,j}^R$

$for(l=1; l\le N\left(n(v_{i,j}^R)\right); l++)$

{

$//v_{i,j,l}^R - l - ая\, вершина\, из\, множества\, n(v_{i,j}^R)$

$//r(v_i^R,v_j) - расстояние\, между\, ИСЭ, соответствующими\, v_i^R\, и\, v_j$

$if\left(r(v_i^R,v_j)\le S(v_i^R)*k \vee m(e(v_{i,j}^R,v_{i,j,l}^R))=1 \vee v_{i,j,l}^R \notin b(v_i^R)\right)$

{

$b(v_i^R)+=v_{i,j,l}^R;$

}

}

}

Выбранные для дальнейшего сравнения вершины соединяются ребрами каждое с каждым. В качестве атрибута данных рёбер выступает расстояние между соответ-

ствующими ИСЭ – $R(e(v_i, v_j))$. Фактически формируется новый граф-клика из репрезентативных вершин, который и используется при дальнейшем сравнении изображений.

Таким образом, при данном сравнении изображений используется только информация, предоставляемая графом-описанием, в отличие от других, которые основаны на использовании сложных дескрипторов.

Сравнение репрезентативных графов

Для поиска соответствующих подграфов у пары графов-описаний проводится их сравнение, основанное на поиске их изоморфных пересечений. Для данного сравнения предлагается использовать построение дополнительного графа-пирамиды [10], [11], поскольку это позволяет достичь линейной зависимости вычислительной сложности от количества решений, а также даёт возможность ускорения вычислений за счёт учёта специфики конкретной задачи и возможность распараллеливания вычислений.

Производный граф

Поиск соответствующих подграфов начинается с построения производного графа. Вершинам данного графа $G^P(V^P, E^P)$, соответствуют пары вершин сравниваемых графов $G^1(V^1, E^1)$ и $G^2(V^2, E^2)$, которые потенциально могут соответствовать друг другу – $(v_i^1, v_j^2) \in V^P \Leftarrow v_i^1 \in V^1, v_j^2 \in V^2, atr(v_i^1) \approx atr(v_j^2)$.

Ребрами в производном графе соединены те вершины, соответствующие вершины которых также являются смежными, и параметры, характеризующие рёбра, совпадают или достаточно похожи друг на друга.

$$e\big((v_i^1, v_j^2), (v_l^1, v_k^2)\big) \in E^P \Leftarrow e(v_i^1, v_l^1) \in E^1, e(v_j^2, v_k^2) \in E^2, atr(e(v_i^1, v_l^1)) \approx atr(e(v_j^2, v_k^2)).$$

Набор параметров, составляющих атрибуты вершин и рёбер, и то, как они сопоставляются, зависит от решаемой задачи. В данном случае сравнение графов используется для поиска одинаковых объектов, поэтому предполагается, что значительных искажений пропорций, отношений яркости и контрастности не происходит. Это позволяет ограничиться тем набором параметров, который указывался ранее – масштабом, яркостью и контрастностью.

В рассматриваемом случае сравнения графов-описаний это означает, что соответствующие им структурные элементы совпадают или достаточно друг на друга похожи. Следует, однако, учитывать, что при изменении освещения значения, характеризующие ИСЭ могут меняться, что приводит к росту количества вершин производного графа.

Для оценки похожести структурных элементов, соответствующих вершинам v_i^1 и v_l^2, используются следующие соотношения:

$$rS(v_i^1, v_j^2) = S(v_i^1) / S(v_j^2),$$

$$rG(v_i^1, v_j^2) = G(v_i^1) / G(v_j^2),$$

$$rDoG(v_i^1, v_j^2) = DoG(v_i^1) / DoG(v_j^2),$$

$$rGDoG(v_i^1, v_j^2) = \frac{G(v_j^1) * DoG(v_i^2)}{DoG(v_j^1) * G(v_i^2)}.$$

Вершина (v_i^1, v_j^2) образовывается только в случае, если удовлетворяется условие

$$\begin{aligned}&1/mrS \le rS(v_i^1, v_j^2) \le mrS \wedge 1/mrG \le rG(v_i^1, v_j^2) \le mrG\\&\wedge 1/mrDoG \le rDoG(v_i^1, v_j^2) \le mrDoG \wedge 1/mrGDoG \le rGDoG(v_i^1, v_j^2) \le mrGDoG\end{aligned}, \quad (1)$$

где mrS, mrG, $mrDoG$, $mrGDoG$ – пороги, определяющие максимальное отношение между соответствующими параметрами атрибутов сравниваемых вершин.

В данном случае это означает, что взаимное расположение на плоскости и в пространстве признаков соответствующих структурных элементов совпадает. Сравнение атрибутов рёбер предназначено для проверки схожести их взаимного пространственного и параметрического расположения.

Для сравнения их взаимного расположения в плоскости изображений используются следующие соотношения:

$$rR(e\left((v_i^1, v_j^2), (v_l^1, v_k^2)\right)) = R(e(v_i^1, v_l^1)) / R(e(v_j^2, v_k^2)),$$

$$rS_1(e\left((v_i^1, v_j^2), (v_l^1, v_k^2)\right)) = S(v_i^1) / R(v_j^2),$$

$$rS_2(e\left((v_i^1, v_j^2), (v_l^1, v_k^2)\right)) = S(v_l^1) / R(v_k^2),$$

$$rS(e\left((v_i^1, v_j^2), (v_l^1, v_k^2)\right)) = (rS_1(e\left((v_i^1, v_j^2), (v_l^1, v_k^2)\right)) + rS_2(e\left((v_i^1, v_j^2), (v_l^1, v_k^2)\right))) / 2,$$

$$drRS(e\left((v_i^1, v_j^2), (v_l^1, v_k^2)\right)) = abs(rR(e\left((v_i^1, v_j^2), (v_l^1, v_k^2)\right)) - rS(e\left((v_i^1, v_j^2), (v_l^1, v_k^2)\right))),$$

$$drRR(e\left((v_i^1, v_j^2), (v_l^1, v_k^2)\right)) = abs(rR_1(e\left((v_i^1, v_j^2), (v_l^1, v_k^2)\right)) - rR(e\left((v_i^1, v_j^2), (v_l^1, v_k^2)\right))).$$

При сравнении взаимных яркостей пар ИСЭ используется качественная оценка. То есть для каждой рассматриваемой пары ИСЭ учитываются только отношения вида «больше», «меньше», «равно». Как показали проведённые исследования, учёт только такого типа отношений позволяет добиться большего качества, чем использование количественных отношений. Для этого используется следующая функция:

$$RG(e(v_i^1, v_l^1)) = \begin{cases} -1, if\ -thrG > rG(e(v_i^1, v_l^1)) \\ 0, if\ -thrG \le rG(e(v_i^1, v_l^1)) \le thrG \\ 1, if\ rG(e(v_i^1, v_l^1)) > thrG \end{cases},$$

где $rG(e(v_i^1, v_l^1)) = \dfrac{G(v_i^1) - G(v_l^1)}{G(v_i^1) + G(v_l^1)}$, $thrG$ – порог, определяющий разницу в яркостях, которым можно пренебречь, чтобы яркости считались одинаковыми.

Структурные особенности изображений слабо подвержены влиянию смены освещения, что позволяет отфильтровывать ложные соответствия при значительных вариациях яркости, контрастности и фона. Для того, чтобы учесть структурные особенности изображений при их сравнении, необходимо сравнить взаимное расположение соседних ИСЭ для каждой из сравниваемых пар.

Для сравнения структур изображений в окрестности ЛИЧ соответствующим вершинам $v_i^1, v_j^2, v_l^1, v_k^2$, используется следующий приём, – исходя из условия перехода данных ЛИЧ друг в друга, рассчитывается преобразование координат $M: v_j^2 = M\vec{r}(v_i^1), v_k^2 = M\vec{r}(v_l^1)$ из одного изображения в другое, используя которое, проводится сравнение расположений

ИСЭ, соответствующих вершинам из множеств $b(v_i^1), b(v_j^2), b(v_l^1), b(v_k^2)$. Для оценки похожести структур используется функция

$$C(v^1, v^2) = \sum_{v_m^1 \in b(v^1)} c(v_m^1, b(v^2), M),$$

где $c(v_m^1, b(v^2), M) = \begin{cases} 1, if\ \exists v_n^2 \in b(v^2): \left|\vec{r}(v_n^2) - M\vec{r}(v_m^1)\right| < mD \\ 0, if\ \neg\exists v_n^2 \in b(v^2): \left|\vec{r}(v_n^2) - M\vec{r}(v_m^1)\right| < mD \end{cases}.$

Если значения $C(v_i^1, v_j^2)$ и $C(v_l^1, v_k^2)$ превышают некий порог mC – $C(v_i^1, v_j^2) > mC$, $C(v_l^1, v_k^2) > mC$, то структуры считаются похожими.

Ребро $e\left((v_i^1, v_j^2), (v_l^1, v_k^2)\right)$ образовывается только в случае соблюдения условия

$$\begin{aligned} &drRS(e\left((v_i^1, v_j^2), (v_l^1, v_k^2)\right)) \le mdrRS \wedge drRR(e\left((v_i^1, v_j^2), (v_l^1, v_k^2)\right)) \le mdrRR \wedge \\ &RG(e(v_i^1, v_l^1)) = RG(e(v_j^2, v_k^2)) \wedge C(v_i^1, v_j^2) > mC \wedge C(v_l^1, v_k^2) > mC \end{aligned}. \quad (2)$$

Следует отметить, что в производном графе могут существовать подмножества вершин, каждая из которых соответствует одной вершине одного графа и различным вершинам второго. Причём вершины второго графа соответствуют структурным элементам, сходным по параметрам и расположению. Это приводит к наличию независимых множеств в производном графе, которое приводит к экспоненциальному количеству максимальных клик. Чтобы этого избежать, достаточно попарно соединить ребрами такие вершины, что позволяет исключить из производного графа независимые множества, обуславливающие экспоненциальное количество максимальных клик.

Построение пирамиды

Пирамида образуется в результате построения, то есть начиная с основного / исходного графа (нулевого уровня) проводится следующая процедура – граф, соответствующий одному уровню, разбивается на подграфы, для каждого из которых в следующем уровне создаётся новая вершина. Эти новые вершины являются дочерними для вершин соответствующих подграфов предыдущего уровня, которые, в свою очередь, являются родителями для новозаведенных вершин.

Вид пирамиды и, соответственно, то, какие подграфы будут соответствовать вершинам пирамиды, будет зависеть от правил её построения. Поиск пресечений графов заключается в следующем – на одном из графов строится пирамида, а затем на производном графе строится вторая пирамида по образу и подобию первой. В соответствии с такими правилами построения вершинам второй пирамиды будут соответствовать пересечения рассматриваемых графов.

Однако в случае графов, описывающих изображения, можно ограничиться построением одной пирамиды. Действительно, если графы G^1 и G^1 являются кликами, то их пересечения также являются кликами. Поэтому для их поиска достаточно найти все максимальные клики в производном графе. Для этого необходимо построить граф-пирамиду для поиска множества максимальных клик.

Для того чтобы вершинам пирамиды соответствовали клики, необходимо, чтобы подграфы разбиения являлись кликами, а новые вершины пирамиды соединялись ребрами только в том случае, если их объединение также является кликой. Если придерживаться данных правил при построении пирамиды, то каждой её верхушке будет соответ-

ствовать клика. Наиболее подходящей для решения данной задачи является ребёрная пирамида, у которой в качестве подграфов разбиения выступают пары смежных вершин. То есть каждой вершине данной пирамиды соответствует ребро с инцидентными ей вершинами.

Существует, однако, несколько дополнительных правил построения такой пирамиды, которые позволяют избегать повторов и пропусков решений. Более подробно о построении пирамиды для поиска множества максимальных клик можно посмотреть в [11].

Сокращение вычислений

Если графы G^1 и G^2 являются кликами, то для создания производного графа может потребоваться значительное время. Это связано с необходимостью создания большого числа рёбер $-O\left(\left(N(V^1)N(V^2)\right)^2\right)$.

Также два таких графа будут иметь большое количество пересечений, не соответствующих одинаковым объектам. Это обусловлено высоким количеством структурных элементов, что приводит к большому числу их подмножеств, имеющих сходное / похожее взаимное расположение. Следует также учесть, что структурные элементы, относящиеся к одному объекту, находятся недалеко друг от друга. Следует помнить также, что вычислительная сложность поиска множества клик в графе методом строительства пирамиды линейна относительно их числа и пропорциональна кубу среднего количества их вершин.

Вышесказанное позволяет заключить, что для сокращения вычислений необходимо уменьшить количество вершин в отдельных максимальных кликах. Это необходимо сделать, выбросив из рассмотрения рёбра сравниваемых графов, соединяющие вершины, которые соответствуют далеко расположенным друг от друга структурным элементам.

Контроль взаимного расположения структурных элементов

Метод построения пирамиды позволяет использовать дополнительные проверки для контроля свойств промежуточных результатов. Это реализуется с помощью дополнительных условий при создании новых вершин и рёбер пирамиды.

При сравнении графов, описывающих изображения, необходимо контролировать взаимное расположение структурных элементов в каждой паре потенциально соответствующих клик-троек.

Перед построением ребра между вершинами, объединению которых соответствует клика-тройка, проводится проверка. Направление обхода по соответствующим структурным элементам данной пары троек должно совпадать, иначе новое ребро не образуется. Это позволяет отфильтровывать зеркально отражённые соответствия.

Также проверяются соотношения между соответствующими расстояниями между центрами ИСЭ. Данные отношения не должны сильно отличаться, что позволяет отфильтровывать геометрически искаженные соответствия.

Формирование решения

После построения пирамиды её вершинам, не имеющим дочерних, будут соответствовать максимальные клики производного графа $\{q_i^P\}$ [11]. Данным кликам отвечают

пересечения – пары соответствующих подграфов, принадлежащих рассматриваемым графам-описаниям G^1 и G^2. Вообще говоря, данные графы не изоморфны, а гомоморфны. Это связано с тем, что в найденных кликах могут быть вершины, соответствующие одной вершине одного графа и нескольким различным вершинам другого, что обусловлено их попарной смежностью. Не смотря на это, данным подграфам соответствуют группы структурных элементов, взаимное расположение которых совпадает с определённой точностью.

Вследствие того, что при сравнении графов были исключены рёбра, соединяющие вершины, которым соответствуют структурные элементы, расположенные далеко друг от друга, найденные пересечения не являются максимальными. Для того, чтобы найти максимальные пересечения, необходимо объединить найденные клики производного графа в подграфы, которым отвечают соответствующие клики графов G^1 и G^2.

Это можно сделать рядом способов, каждый из которых лучше подходит к различным случаям. Если необходимо получить как можно более точно решение – т.е. чтобы взаимное расположение соответствующих друг другу структурных элементов полностью совпадало, необходимо провести следующую процедуру. На основе анализа уже известных пересечений удалить из производного графа вершины, которые не соответствуют искомым решениям. Затем дополнить производный граф рёбрами, которые соответствуют далеко удаленным друг от друга парам структурных объектов. После этого построить новый граф G_N, каждой вершине которого отвечает найденная пара соответствующих подграфов, т.е. одна из максимальных клик производного графа: $v_i^N = v(q_i^P)$. Ребрами соединить только те вершины, которым соответствуют подграфы производного графа, объединение которых является кликой. После чего производится поиск множества максимальных клик в данном графе G_N. Согласно заданным критериям из найденного множества выделяются те клики, которые соответствуют искомым решениям.

Однако если учитывать, что вследствие различного рода искажений взаимное расположение некоторых структурных элементов может меняться, то условие создания рёбер в графе G_N необходимо ослабить. Например, потребовать, чтобы количество несопряженных пар в объединённом подграфе было не более определённого порога. А после создания всех рёбер графа G_N действовать так же, как и в первом случае.

Следует, однако, отметить, что создание новых рёбер и проверка каждой пары клик в производном графе требует дополнительного времени. Уменьшить временные затраты позволяет использование того свойства соответствующих графов, что взаимное расположение их структурных элементов совпадают (или достаточно близко). Это означает, что координаты структурных элементов рассматриваемой пары изображений связаны друг с другом определённым преобразованием $M(q_i^P)$. Для двух пар соответствующих множеств структурных элементов, относящихся к одному объекту, данное преобразование координат будет с определённой точностью совпадать: $e(v_i^N, v_j^N) \Leftrightarrow M(q_i^P) \approx M(q_j^P)$.

В данном случае преобразование $M(q_i^P)$ находилось с помощью метода наименьших квадратов из условия отсутствия аффинных искажений, т.е. преобразование подразумевает только трансляцию, поворот и масштабирование. Сравнение преобразований $M(q_i^P)$ и $M(q_j^P)$ сводится, таким образом, к сопоставлению трансляций

$\vec{t}(M)$, углов поворота $\alpha(M)$ и масштабных коэффициентов $c(M)$. Для их сравнения использовались следующие величины:

$$dt(q_i^P, q_j^P) = \left|\vec{t}(M(q_i^P)) - \vec{t}(M(q_j^P))\right|,$$

$$d\alpha(q_i^P, q_j^P) = \left|\alpha(M(q_i^P)) - \alpha(M(q_j^P))\right|,$$

$$rc(q_i^P, q_j^P) = \frac{\max(c(M(q_i^P)), c(M(q_j^P)))}{\min(c(M(q_i^P)), c(M(q_j^P)))}.$$

Преобразования $M(q_i^P)$ и $M(q_j^P)$ считаются совпадающими, если соблюдается условие

$$dt(q_i^P, q_j^P) \le mdt \wedge d\alpha(q_i^P, q_j^P) \le md\alpha \wedge rc(q_i^P, q_j^P) \le mrc. \qquad (3)$$

Таким образом, каждой найденной максимальной клике производного графа вычисляется и ставится в соответствие преобразование координат. Если пара таких клик имеет совпадающее преобразование, то вершины графа G_N соединяются ребром.

Для ускорения построения ребер в графе G_N данный процесс проводится в несколько этапов – вначале рассматриваются клики, имеющие общие вершины; затем выделяются связные подграфы в G_N; затем рассматриваются пропущенные пары, принадлежащие одному связному подграфу. Данный приём позволяет значительно сократить время, по сравнению с полным перебором всех пар найденных максимальных клик (всех пар вершин графа G_N).

Если объект на одном изображении довольно значительно искажен по сравнению со вторым изображением, то преобразования координат будут совпадать только у максимальных клик производного графа, имеющих общие вершины. В этом случае целесообразно остановится на этапе выделения связных подграфов и рассматривать соответствующие им подграфы производного графа как возможные кандидаты решения. Искомые решения выбираются из найденных кандидатов согласно заданным критериям, зависящим от конкретной задачи.

Эксперимент

Для проверки эффективности использования предложенных графов и средств их обработки (далее ГиС) были выбраны задачи поиска объекта на изображении и поиска одинаковых объектов на различных изображениях.

Для проверки работоспособности представленного алгоритма на основе использования ГиС вначале поиск объекта проводился на искусственно сформированных изображениях. Данные изображения формировались следующим образом – на произвольное изображение, которое служит фоном, накладывалось изображение объекта, изменённое в размерах и повёрнутое на произвольный / лучайный угол. Место расположения изображения объекта на фоне также выбиралось случайным образом. Были использованы два типа объектов – текстурированные и слабо текстурированные. В качестве вторых были использованы символы, использующиеся в автомобильных номерных знаках. На рис. 1 представлены изображения объектов, которые были использованы в экспериментах. На рис. 2 представлены изображения, которые применялись в качестве фона.



a



b

Рисунок 1 – Изображения объектов – a – текстурированные объекты,
b – слаботекстурированные объекты



Рисунок 2 – Изображения фона

Граф, представляющий изображение объекта, является кликой, вершинам которой соответствуют ЛИЧ. Граф изображения фон + объект состоит из пересекающихся клик, вершины которых соответствуют ЛИЧ, расположенных друг относительно друга на расстоянии не больше, чем $\left(S(v_i^I)+S(v_j^I)\right)\cdot 5$. Каждой ЛИЧ общих графов ставится в соответствие множество ИСЭ, характеризующих структуру в её окрестности и отстоящих от её центра на расстояние не более чем $S(v_i^I)\cdot 2,5$.

При построении вершин производного графа учитываются яркость, контрастность и отношения яркости и контрастности.

В условии (1) для образования вершин производного графа были использованы параметры $mrS = 2, mrG = 1,5, mrDoG = 1,2,\ mrGDoG = 1,2$. В условии (2) для образования рёбер производного графа – $mdrRS = 0,2, mdrRR = 0,2, mC = 0,5$. В условии (3) для образования рёбер в графе G_N – $mdt = 1,3, md\alpha = 0,07,\ mrc = 0,05$. Такой набор параметров позволяет сократить размеры производного графа и, соответственно, сократить время поиска до 1сек. Было протестировано 1000 изображений, образованных текстурированными объектами. Доля верно обнаруженных объектов составила 97%.

Этот же набор параметров был использован в задачах поиска объекта на изображениях реальной видеопоследовательности и поиска одинаковых объектов на различных изображениях. Поиск одинаковых объектов осуществлялся на искусственно сформированных изображениях. Для каждой из этих задач также было протестировано по 1000 изображений.

Результаты тестов представлены в табл. 1. Из данной таблицы видно, что эффективность предложенных средств на реальных изображениях не сильно падает по сравнению с искусственными и остаётся на достаточно высоком уровне. Это показывает, что результаты тестов на искусственных изображениях позволяют оценивать эффективность предложенных средств при их использования для реальных изображений.

Таблица 1 – Результаты тестирования на различных задачах

Задача	Доля верного обнаружения объекта
Поиск объекта на искусственно сформированных изображениях	97%
Поиск объекта на изображениях реального видеоряда	95%
Поиск одинаковых объектов на различных изображениях без наличия априорной информации о них	95%

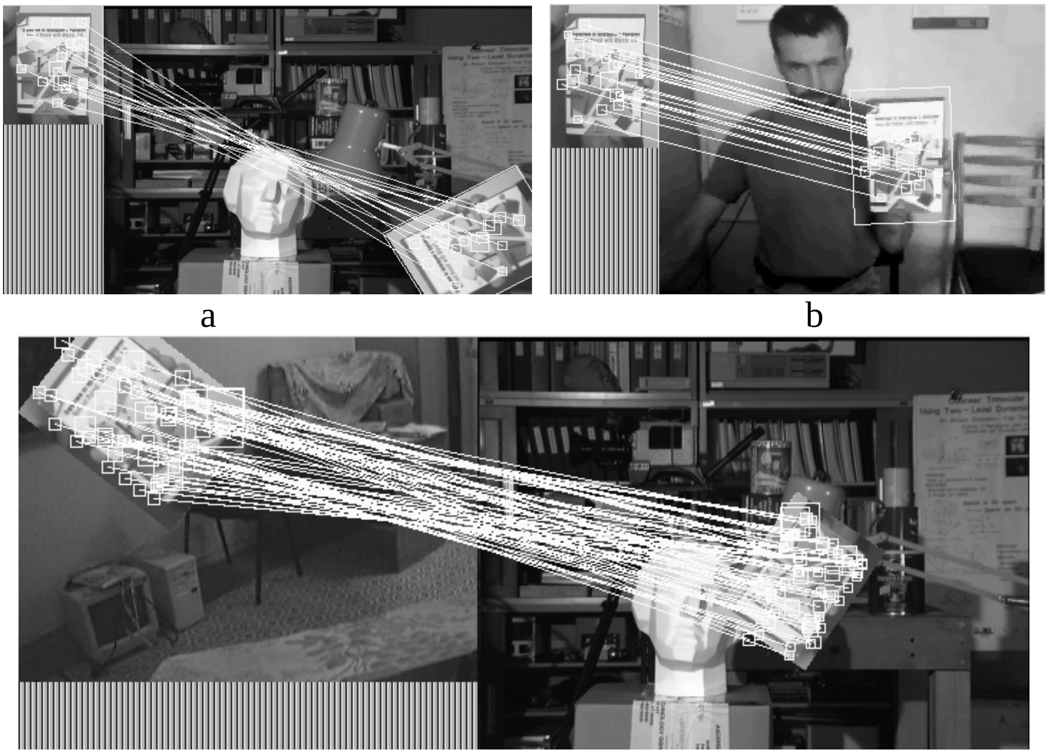
На рис. 3 представлены примеры поиска объекта на искусственном (3a), реальном (3b) изображениях и поиска одинаковых объектов на различных изображениях (3c).

Также было проведено сравнение предложенных средств с аналогами, предназначенными для поиска объектов на изображениях. В качестве аналогов были использованы схемы SIFT+FLANN+RANSAC и SURF+FLANN+RANSAC, которые нашли широкое применение при решении задач распознавания изображений и поиска объектов. Данные схемы были реализованы с помощью библиотеки OpenCV.

В этом тесте фактор времени (скорости получения результатов) не учитывался, чтобы получить как можно более высокий результат как для аналогов, так и для предложенных средств. В условии (1) для образования вершин производного графа были использованы параметры $mrS = \infty, mrG = 2, mrDoG = 2, mrGDoG = 2$. В условии (2) для образования рёбер производного графа – $mdrRS = 0,2, mdrRR = 0,2, mC = 0,5$. В условии (3) для образования рёбер в графе G_N – $mdt = 1,3, md\alpha = 0,07, mrc = 0,05$.

Было протестировано 3000 изображений. Результаты тестов применения рассматриваемых схем для решения различных задач представлены в табл. 2. Из данной таблицы видно, что доля верно найденных / обнаруженных текстурированных объектов для сравниваемых схем / технологий / механизмов практически не отличается. Для предложенных средств доля изображений, на которых объект был обнаружен верно, составила 99,5% для текстурированных объектов и 98,4% – для слабо текстурированных. Однако для слабо текстурированных объектов контрольные схемы / технологии / механизмы показывают долю верных обнаружений менее 50%, что говорит об их неприменимости в подобных случаях.

Также следует отметить, что метод RANSAC невозможно применять для поиска одинаковых объектов на различных изображениях, поскольку он предназначен для поиска объекта по известному шаблону. Это отражено в табл. 2 прочерками в соответствующих ячейках.



a b c

Рисунок 3 – Примеры поиска объекта – a – на искусственном изображении, b – на реальном изображении, c – одинаковых объектов на различных изображениях

Выводы

Применение представленных графов (репрезентативный граф и граф-описание) и средств их обработки позволяет эффективно проводить поиск текстурированных и слабо текстурированных объектов на изображениях, в отличие от других методов и алгоритмов, которые специализируются на объектах одного типа. При этом используются только данные, предоставляемые графом-описанием, без использования дополнительных дескрипторов. Это показывает высокую перспективность использования предложенных средств формирования и обработки описаний изображений с помощью графов для решения задач распознавания зрительных образов.

Таблица 2 – Результаты сравнительных тестов – доля правильно найденных объектов в зависимости от типа объекта, задачи и используемых средств

	Поиск объекта на искусственно сформированных изображениях		Поиск одинаковых объектов на различных изображениях без наличия априорной информации о них	
	Текстурированные объекты	Нетекстурированные объекты	Текстурированные объекты	Нетекстурированные объекты
SIFT+FLANN +RANSAC	99%	44,1%	-	-
SURF+FLANN +RANSAC	99%	46,3%	-	-
ГиС	99,5%	98,4%	99%	97%

Их применение также даёт возможность увеличивать скорость работы за счет распараллеливания вычислений, которое легко осуществимо вследствие особенностей структуры используемых алгоритмов.

Дальнейшие исследования в области применения представленных графов и средств их обработки будут направлены на увеличение скорости вычислений и расширение круга решаемых задач. Увеличение скорости работы возможно за счет распараллеливания вычислений, которое легко осуществимо вследствие особенностей структуры используемых алгоритмов.

Учет структуры изображения в окрестности структурных элементов, которые соответствуют репрезентативным вершинам, позволит лучше отфильтровывать ложные гипотезы при построении производного графа. Это позволит сократить его размеры, что уменьшит время получения решения. Исследования в этой области также позволит более оптимально для решаемых задач выбирать вершины для репрезентативного графа, что позволит добиться стабильно высокой эффективности для всех типов объектов.

Литература

1. Martin A. Fischler. Random Sample Consensus: A Paradigm for Model Fitting with Apphcatlons to Image Analysis and Automated Cartography / Martin A. Fischler, Robert C. Bolles // Communications of the ACM. - June 1981. – Vol. 24, № 6. – P. 381-395
2. Krystian Mikolajczyk. A performance evaluation of local descriptors / Krystian Mikolajczyk, Cordelia Schmid // IEEE Transactions on Pattern Analysis & Machine Intelligence. – 2005. – Vol. 27, № 10. – P. 1615-1630.
3. Yan Ke. PCA-SIFT: A More Distinctive Representation for Local Image Descriptors / Yan Ke, Rahul Sukthankar // IEEE Computer Society Conference on Computer Vision and Pattern Recognition. – 2004. – V. 2. – P. 506-513.
4. Fred Rothganger. 3D Object Modeling and Recognition Using Local Affine-Invariant Image Descriptors and Multi-View Spatial Constraints / Fred Rothganger, Svetlana Lazebnik, Cordelia Schmid // International Journal of Computer Vision archive. – Vol. 66. – P. 231-259.
5. Benoit Huet. Shape retrieval by inexact graph matching / Benoit Huet, Andrew D.J. Cross, Edwin R. Hancock // International Conference on Multimedia Computing and Systems. – 1999. – Vol. 1. – P. 772-776.
6. Radu Horaud. Stereo Correspondence Through Feature Grouping and Maximal Cliques / Radu Horaud, Thomas Skordas // IEEE Transactions on pattern analysis and machine intelligence. – 1989. – Vol. 11, № 11. – P. 1168-1180
7. Агарков А.В. Ієрархічне представлення зображень для застосування при вирішенні широкого кола проблем / А.В. Агарков // Оброблення сигналів і зображень та розпізнавання образів : Восьма всеукраїнська міжнародна конференція. - Київ, 2006. – С. 119-122.
8. Агарков А.В. Структурное описание изображений в виде графа для решения задач распознавания / А.В. Агарков // Бионика интеллекта. – 2009. – № 1(70). – С. 95-101.
9. Andrej Agarkov. Novel Structural Elements for Image Descriptor / Andrej Agarkov // International Workshop on Intelligent Information Systems. – Institute of Mathematics and Computer Science. – September 13 – 14, 2011. – P. 97-100
10. Агарков А.В. Метод сравнения двух графов за полиномиальное время / А.В. Агарков // Искусственный интеллект. – 2003. – № 4. – С.172-184.
11. Агарков А.В. Поиск множества максимальных клик на основе метода построения дополнительного графа / А.В. Агарков // Искусственный интеллект. – 2011. – № 3. – С .190-199.

Literatura

1. Martin A. Fischler, Robert C. Bolles, Random Sample Consensus: A Paradigm for Model Fitting with Apphcatlons to Image Analysis and Automated Cartography // Communications of the ACM, - June 1981. – vol. 24, № 6. - P. 381-395
2. Krystian Mikolajczyk and Cordelia Schmid, A performance evaluation of local descriptors // IEEE Transactions on Pattern Analysis & Machine Intelligence. - 2005. – Vol. 27, № 10. – P.1615–1630.
3. Yan Ke, Rahul Sukthankar. PCA-SIFT: A More Distinctive Representation for Local Image Descriptors // IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2004. Vol.2. – P.506-513.
4. Fred Rothganger, Svetlana Lazebnik, Cordelia Schmid, 3D Object Modeling and Recognition Using Local Affine-Invariant Image Descriptors and Multi-View Spatial Constraints, International Journal of Computer Vision archive. Vol. 66. P.231 – 259.

5. Benoit Huet, Andrew D. J. Cross, Edwin R. Hancock, Shape retrieval by inexact graph matching // International Conference on Multimedia Computing and Systems. 1999. Vol. 1. P.772-776.
6. Radu Horaud, Thomas Skordas, Stereo Correspondence Through Feature Grouping and Maximal Cliques // IEEE Transactions on pattern analysis and machine intelligence, november 1989. vol. 11, №11, P. 1168-1180.
7. Agarkov A.V. Obroblennja sygnaliv i zobrazhen’ ta rozpiznavannja obraziv: Vos’ma vseukrains’ka mizhnarodna konferencija. Kyiv. 2006. S. 119-122.
8. Agarkov A.V. Bionika intellekta. 2009. №.1(70). S. 95-101.
9. Andrej Agarkov, Novel Structural Elements for Image Descriptor // International Workshop on Intelligent Information Systems, Institute of Mathematics and Computer Science, - September 13-14, 2011. P. 97-100.
10. Agarkov A.V. Iskusstvennyj intellekt. 2003. № 4. S. 172-184.
11. Agarkov A.V. Iskusstvennyj intellekt. 2011. № 3. S. 190-199.

RESUME

A.V. Agarkov

Objects Searching on Images by Using Graphs

This work is devoted to finding objects in the image. To solve this problem are encouraged to use a method based on the use of a graph that describes the image. This graph should have the property that the individual objects in the image must comply with its subgraphs. Then the problem of finding the object is reduced to an appropriate allocation of the subgraph.

To describe the images suggested to use a graph, constructed on the basis of the structural elements identified with the help of multi-scale representation. The same multi-scale representation constructed on the basis of the convolution operator DoG, is used to find local invariant features in images, which correspond to local extrema. But unlike them, these structural elements are obtained by analyzing the extrema in the direction in all planes of multi-scale representation.

This graph can reflect the structure of both textured and non-textured objects. It allows you to build the skeleton of regions with uniform brightness, without prior segmentation of the image.

To search for items using a graph-clique, for which the representative vertexes are elected from of the graph-description. Using this graph instead of graph-descriptions can reduce search time. Each vertex of the graph is associated with the set of adjacent structural elements, which characterizes the structure of the image in its vicinity.

Selecting objects in the images is reduced to compared of the corresponding graphs. If one image is an image of the object, it looks for a particular object. If it’s pictures of various scenes, it looks for the same objects.

To separate subgraphs corresponding to the desired objects, tools are used, based on the construction of additional graph pyramid. Pre-complementary graph is constructed on the basis of the compared graphs. Cliques of the one correspond to the intersection of the compared graphs. To select the set of maximal cliques algorithm based on the construction of an additional graph pyramid.

Experimental verification of a given graph, and means of processing to search for textured and weakly textured objects demonstrated their high efficiency. Comparison with analogues showed a significant superiority of the proposed means of finding poorly textured objects.

Статья поступила в редакцию 05.06.2012.